

# Turing and ACE Lessons from a 1946 Computer Design

*B.E. Carpenter*

CN Division, CERN, Geneva, Switzerland

## Abstract

In March 1946, Alan Turing, best known as a British pure mathematician, presented the world's first complete design for a stored-program electronic computer, ACE. Although Turing had seen the draft report on EDVAC by von Neumann, the ACE design was very different and included detailed circuit diagrams as well as software examples, and a precise budget estimate. How did a theoretician manage this? What happened to the design? What happened to Turing? Why did British industry not build on this early start? And has anything really fundamental been invented since 1946?

## 1 A Design out of Nowhere

Alan Turing was not famous in the first post-war year of 1946, but he was known to a small circle of theoretical mathematicians as one of the two people who, in 1937, had published independent solutions to the “decidability problem”. This was the question, posed by David Hilbert, whether mathematics was in principle decidable, i.e. was there a guaranteed method of testing the truth or falsity of any well-formulated mathematical proposition. Turing, and Alonzo Church, both proved that there is, disappointingly, no such method. Church, who was at Princeton, invented a rather neat proof using his own lambda-calculus notation for recursive function theory. Turing, at Cambridge, invented a proof based on the notion of a computable number and a notional computing machine, nowadays known as a Turing machine. He showed that some computations could be defined which would never terminate, and this sufficed to prove the undecidability result. It would be hard to imagine any piece of theoretical work more removed from practical applications in 1937.

Turing went to Princeton to work with Church, where he met John von Neumann, and also gained a doctorate, before returning to Cambridge. World War II started soon afterwards, and even Turing's family did not know what he was doing throughout the war, although they knew he was not in the armed forces. Immediately after the war ended he took a job at the National Physical Laboratory (NPL) near London. In March 1946 he presented a report to the NPL Executive Committee entitled “Proposal for Development in the Mathematics Division of an Automatic Computing Engine (ACE).” It was a detailed report explaining the basic principles of stored-program digital computers, describing the architecture and instruction set of such a computer, giving circuit diagrams for all the important circuits, several software examples, and a budget estimate of £11,200 (about twenty times Turing's salary at the time). Only the wiring lists and mechanical drawings were lacking.

At that time, the military world had three major computational problems to solve – decryption of military signals, computation of ballistic tables, and simulation of atomic bomb explosions. As far as public knowledge went, such tasks could be tackled only by rooms full of tens or hundreds of operators of desk calculators – such people were known as “computers” as it happened. Here was an academic, an “egghead”, in fact a man with a shy and difficult personality, offering what seemed to be a serious and revolutionary approach. Where did this design come from?

It did not, of course, come entirely out of Turing's head. However, he explicitly

documented the fact that ACE was logically equivalent to the Universal Turing Machine, the special case of his 1937 machines which could compute any computable number. Indeed an infinite loop on ACE, and every other computer, is just an attempt to compute an uncomputable number. All programmers have proved the undecidability of mathematics many times! There were many other threads leading to the ACE design, too. The flip-flop circuit and the and-gate circuit were both invented by 1925, actually for use in cosmic ray detectors, early examples of technology fallout from high energy physics. The notions of storing digital values electronically, and of computing electronically, were invented by John Atanasoff in Iowa in the late 1930's. (German readers must excuse the absence of a discussion of the equally pioneering work of Konrad Zuse, which had no influence on Anglo-American developments.) Recirculating memory based on sound waves in tanks of mercury was invented by British radar developers during World War II. The engineering of large digital systems had been analysed by Shannon in 1938, and the basic idea of a stored-program computer was developed from Atanasoff's ideas by Eckert and Mauchly during the war. Their proposal for a computer called EDVAC was written up by von Neumann (who knew Turing while the latter was gaining his Ph.D. at Princeton before the war) in 1945, but it was at this stage only an outline compared to Turing's subsequent ACE report. EDVAC and ACE shared the key idea: programs and data were to be stored in the same memory.

The fact is that during the war, Turing was one of the critical participants in the decryption work which was at the heart of all significant British intelligence gathering. By intercepting encrypted military signals, decrypting them, and making careful use of the results, the British were able, for example, to route shipping away from German submarine attacks. This was of vital importance at the time when Britain's only external source of food or military equipment was from transatlantic shipping. British success in deciphering German signals, and the linked American success with Japanese signals, was a closely guarded secret until the 1970s, long after Turing was dead. He invented some of the basic methods used and actively led one of the actual decryption teams. In this work he saw, helped to design, and helped to use the first specialised digital electromechanical computing devices – not stored-program devices, therefore not equivalent to the Universal Turing Machine, but not far off. He personally designed and built speech encryption equipment, and he was familiar with COLOSSUS, the first major specialised digital electronic computing device. He also travelled to the USA during the war, for exchange of information about encryption and decryption, and saw von Neumann's EDVAC report almost as soon as it was written. Turing instantly grasped the dramatic potential of digital computers, as did his boss at NPL – and Turing was given the job of writing a proposal for a British rival to EDVAC.

## **2 The original ACE design**

Turing documented the ACE design in an NPL internal report formally published only in 1986 (see Bibliography). In 1945 the only conceivable approach, on financial grounds, was a serial computer design, in which all the bits in a word flow along the same wire and through the same logic gates in sequence. (Incidentally, none of the words "computer", "bit", "word" and "gate" had their modern significance at the time. In this paper I use modern terminology in general.) The least significant bit arrived first, a prerequisite for building a serial adder. The machine was to run with a 1 MHz clock, i.e. to handle one bit every microsecond. ACE was designed with 32 bit words, named "minor cycles" because one word could be stored in a small recirculating memory with a cycle time of 32  $\mu$ sec. The report included a section of applied physics calculations showing how to build mercury delay lines with adequate temperature stability. ACE was to include 32 such one-word temporary storage registers, named TS1 to TS32.

Main memory was to be a set of "major cycles", i.e. 1024-bit delay lines, each therefore containing 32 words. A 15-bit address space was planned, allowing theoretically for

a 32 kW or 128 kB main memory, which would have been immense at the time. Consecutive ACE instructions could not process consecutive words from main memory, since some dead time was needed between instructions whereas the recirculating memory could not be held up. This was overcome by interleaved addressing: words 1, 2, 3,... 16 in a given delay line were actually in physical positions 1, 3, 5,... 31, with words 17 to 32 taking up the other positions. This smart idea was later dropped by NPL, leading to a lot of software gymnastics to optimise performance.

EDVAC had an accumulator-based instruction set such as was found in almost all computers until the 1970's. ACE was different; its instructions were typically register-to-register or register-memory instructions. The ACE program counter, misnamed as CD ("current data"), and the instruction register CI ("current instruction") were the only registers in addition to TS1-32. ACE had only eleven instructions: a branch instruction, five "move" instructions, two input-output instructions, a logical instruction, an arithmetic instruction, and a general-purpose instruction. The last three instructions were "horizontally microcoded" in the sense that various options within the instruction could be combined by setting various bits.

All this was exciting stuff in 1946. However, two other sections of the report are what make it quite remarkable. Firstly, Turing gave a list of ten possible applications for ACE: six in numerical analysis, an analogue application to show what ACE could not do, a personnel management application (counting butchers in the Army) to show what he thought ACE should not do, and two artificial intelligence applications (solving jigsaw puzzles and playing chess). He wrote "by following up this aspect [allowing it to make occasional mistakes] the machine could probably be made to play very good chess." He did not predict that computers would reach Grand Master standard within fifty years, however.

The second remarkable section is about software, headed "Examples of Instruction Tables." *Machine form* (binary) is distinguished from *permanent form* (relocatable code) and from *popular form* (essentially assembly language). There is a clear concept of a subroutine library with several examples of subroutines, e.g. INDEXIN to access an array element. The most surprising subroutines are BURY and UNBURY, which push and pop a subroutine's return address to and from a stack whose pointer is held in register TS31. Inventing this concept in late 1945 was a truly amazing achievement, perhaps inspired by the recursive function theory which Turing had learnt from the work of Church, and by a slight knowledge of the nineteenth century work of Babbage.

The section on software concludes with a program called CALPOL to compute polynomial expressions. It calls a number of the previously defined subroutines, and, needless to say, contains a couple of bugs - probably the first ever recorded for an electronic stored-program computer.

The sheer originality of much of this in 1945-46 cannot be overstated. Although the basic stored-program concept was apparently not Turing's, and the first overall computer design, EDVAC, was American, ACE was the first complete electronic computer design; apart from bugs, it could have been built. The architecture was new and the discussion of software was unparalleled. When the ACE report was made available by NPL in the 1970's it was possible to identify at least fifteen formative ideas in it which could not be found in the EDVAC report written nine months earlier.

ACE, as designed by Turing, was the first register machine, the first machine with logical as well as arithmetic instructions, the first machine with symbolic addresses and readable software, the first machine with a library, floating point, and a stack. ACE should have bootstrapped the British computer industry. It was never built as designed; what was built was hardly a commercial success; and the company that built it ended up, after many changes, as part of Fujitsu. What went wrong?

### 3 ACE and Turing after March 1946

During the war, Turing and his colleagues in the decryption business were given the highest possible priority, even above that of weapons production. Turing had personal access to the Prime Minister, Winston Churchill, when short of staff or otherwise obstructed. Any piece of electronics that Turing could specify would rapidly be built for him. At NPL in 1946, in stark contrast, he had to fight for resources for the ACE project in the normal conditions of a peacetime research laboratory, worsened by post-war austerity. As a pure mathematician with some relatively amateur experience of electronic engineering, he found this extremely frustrating. After a while, Turing returned briefly to Cambridge and then moved to Manchester University where he worked on software and applications for the early Manchester computers. He was perhaps attracted by the fact that the first, very simple, stored-program computer in the world ran at Manchester in June 1948. His life ended tragically in 1954 at the age of 41, following a judicial persecution over his homosexuality. This shocking story was little known until the definitive biography of Turing was published in 1983, followed in 1986 by a successful West End play based on his life.

At NPL, the full ACE project was replaced by the “Pilot ACE” project. Pilot ACE first ran, after many changes of plan and personnel, in May 1950. It was a register machine, like the original design, but instead of the interleaved memory trick and the instruction counter, each instruction specified the address of the next instruction. Programmers were supposed to ensure that each instruction just happened to be coming out of a delay line when it was needed, leading to the only computer design in which the speed of sound in mercury determined the optimum coding! But Pilot ACE worked, and was in fact used to simulate the original design of the CERN Proton Synchrotron in 1955.

The NPL went on to construct a full-size ACE by 1957, but it was obsolete before it was complete, since much better techniques than mercury delay lines were by then available. At the same time, the English Electric Company industrialised the Pilot ACE design and sold it under the name of DEUCE from early 1955 until 1962. Just over 30 machines were sold, including at least one in Australia which was used by Gordon Bell, later the designer of the DEC PDP-11, the first widely known register machine. In the same timeframe IBM sold more than 100 machines (mainly model 1401) in the British market. At that time ICT (the predecessor of ICL) was mainly selling their very simple model 1201, so DEUCE was the flagship of the British computer industry.

People who programmed DEUCE retain affection for it today. But instead of building on its success with DEUCE and their other home-made design, the KDF9, the first machine with a hardware stack, English Electric (renamed English Electric Leo Marconi after several mergers) put their money on RCA clones of the IBM 360 architecture in the mid-1960s. At the same time Ferranti managed to sell fewer than ten copies of Atlas, the world’s first virtual memory machine designed in conjunction with Manchester University. ICT built a market for their 1900 series machines, based on a design bought from Ferranti of Canada. Eventually ICT, English Electric Leo Marconi, and part of Ferranti merged into ICL, with a horribly complicated range of architectures and operating systems to support. Meanwhile, IBM had invented compatibility and DEC had invented the minicomputer market. The story ended only recently, when Fujitsu bought ICL.

What went wrong with the British computer industry? Turing’s genius, and that of many other equally worthy pioneers, shows that it was not a lack of ideas or creativity. It was not a lack of money, at least in the early years when computers were rare and very expensive. There is a school of thought that Government intervention hindered, rather than helped, the industry – it was the Government that brought about the creation of ICL with its over-complicated product line, rather than letting some of the component companies go bankrupt. At an earlier stage, Government funding of computer developments tended to protect weak companies rather than to stimulate innovative risk-taking. Thomas J. Watson and Ken Olsen had no help from the US Government when they brought IBM

and DEC into the computer business. Nor did the various American computer companies which went out of business along the way.

#### **4 Has anything fundamental been invented since 1946?**

One can argue that the EDVAC and ACE designs between them contained all the basic ideas of “von Neumann” computer architecture. This argument treats pipelining, parallelism, cache, virtual memory and vector processing as superficial rather than basic. Turing’s mathematical brain would certainly have taken this view, but he would undoubtedly have been fascinated by ideas outside the mainstream such as dataflow machines and neural networks.

One can also argue that the recent RISC revolution is superficial, in the theoretical sense, especially remembering the eleven simple ACE instructions designed in late 1945. Turing explicitly said he had “often simplified the circuit at the expense of the code” which is surely the basic principle of any RISC architecture.

The revolutions since 1946 have been technological rather than fundamental, as semi-conductors have enabled engineers to do things that Turing and his contemporaries could not even dream of. You can buy a lot of computer for £11,200 today.

### **5 Bibliography**

#### **5.1 Books**

A.M.Turing’s ACE Report of 1946 and Other Papers, edited by B.E.Carpenter and R.W.Doran, MIT Press, 1986, ISBN 0-262-03114-0.

Alan Turing: the Enigma, by Andrew Hodges, Burnett Books, 1983, ISBN 0-09-152130-0 (republished in paperback as Alan Turing: the Enigma of Intelligence).

Early British Computers, by Simon Lavington, Manchester University Press, 1980, ISBN 0-7190-0803-4 (paperback 0-7190-0810-7).

Innovating for Failure: Government Policy and the Early British Computer Industry, by John Hendry, MIT Press, 1989, ISBN 0-262-08187-3.

#### **5.2 Papers**

The other Turing machine, B.E.Carpenter and R.W.Doran, Computer Journal Vol. 20 No. 3, 1977, 269-279.

Programming the Pilot ACE: Early Programming Activity at the National Physical Laboratory, M.Campbell-Kelly, Annals of the History of Computing Vol.3 No.2, 1981, 133-162.

#### **5.3 Play**

Breaking the Code, by Hugh Whitemore, first performed at the Haymarket Theatre, London, October 1986, with Derek Jacobi playing the part of Alan Turing.